ExoCross: a general program for generating spectra from molecular line lists
User's manual

Sergei N. Yurchenko[a], Ahmed Al-Refaie[a], Jonathan Tennyson[a]

[a]Department of Physics & Astronomy, University College London, Gower Street, London WC1E 6BT, United Kingdom

Abstract

ExoCross is a Fortran 95 code for generating spectra (emission, absorption) and thermodynamic properties (partition function, specific heat etc) from molecular line lists.

1. Compilation

ExoCross has no dependacies with any libraries and only requires a compiler that supports Fortran 2003 standard. The makefile flags that can be modified are:

`FORT`: The Fortran compiler

`FFLAGS`: flags passed into the `FORT` compiler

The default flags are for the Intel Compilers with aggressive optimization. A single `make` should compile an executable named `xcross.exe`.

2. Input file structure

ExoCross runs from the command line and uses as input a plain text input file; ExoCross is run with a command of the type
`./xcross.exe < input.inp > output.txt` Often the stack limitations have to be lifted by, for example,

```
ulimit -s unlimited
```

Email addresses: `s.yurchenko@ucl.ac.uk` (Sergei N. Yurchenko), `al-refaie.12@ucl.ac.uk` (Ahmed Al-Refaie), `j.tennyson@ucl.ac.uk` (Jonathan Tennyson)

The ExoCross program is OpenMP parallelized and therefore the number of threads needs to be defined beforehand, e.g.

```
export OMP_NUMTHREADS=4
```

ExoCross uses the Fortran 90 input parsing module input.f90 supplied by Anthony J. Stone [1]. The input file is organized in self-contained input lines or input sections. Below is a generic example of an ExoCross input file for generating cross sections using a Gaussian line profiles with a half-width at half-maximum (HWHM) of 0.5 cm^{-1} from an ExoMol list for NaH on a grid of 25001 wavenumber points ranging from 0 to 25000.0 cm^{-1}, for $T = 296$ K (see file NaH_gauss_298K_0.5.inp).

```
Temperature  296.0
Range 0.0  25000.0

Npoints 25001

Absorption

Gaussian

HWHM 0.5 (cm-1)

Output NaH_abs_gauss_0.5

States 23Na-1H__Rivlin.states

Transitions 23Na-1H__Rivlin.trans
```

The two-column cross section output will be printed into the file NaH_abs_gauss_0.5.xsec, in the following form

```
0.00000000E+00    5.85966901E-57
1.00000000E+00    3.75837740E-55
2.00000000E+00    1.43265206E-53
3.00000000E+00    6.19824797E-52
4.00000000E+00    2.51197771E-48
5.00000000E+00    5.64430375E-41
6.00000000E+00    1.21683891E-32
7.00000000E+00    3.25558472E-26
8.00000000E+00    5.45322731E-22
..............    ..............
..............    ..............
```

In this example `Temperature  296.0` \verb specifies the temperature in K, while the transition files .trans is specified with the keyword Transitions.

The input is not case sensitive, so temperature, Temperature, TEMPERATURE or any other combinations of uppercase and lowercase letters work in exactly the same way. A comma or a space can all be used as delimiters, so, e.g., one can write

```
Range  0, 25000
```

or

```
Range   0 25000
```

to specify the wavenumber range (cm^{-1}). Sometimes keywords have several aliases, which are all equivalent. Lines delimited by parentheses (i.e., round brackets) are ignored and can be used for comments. If in the input there is a line with one of the keyword END, STOP or FINISH all lines after it are ignored. Empty lines cannot be used inside sections, while anywhere else they have no effect and ignored by the program. A complete list of keywords is given in Section 6.

The grid size is defined by

$$\Delta\tilde{\nu} = \frac{\tilde{\nu}_{range}}{N_{points} + 1}.$$

Number of grid points N_{points} (usually an odd number) is defined by the keywords Npoints or Number-of-points:

```
Number-of-points 100001
```

Below is a similar example, where a Doppler line profile is used to simulate an emission cross section spectrum of the CH_4 molecule (see {CH4_Doppler_T1500K_0-1000.inp}).

```
Temperature  1500.0
Range 0.0    1000.0
Npoints  10001

emission
doppler
mass 16.0313

abundance 0.97

output dop_emiss_1500

States  12C-1H4__YT10to10.states

Transitions
  12C-1H4__YT10to10__00000-00100.trans
  12C-1H4__YT10to10__00100-00200.trans
  12C-1H4__YT10to10__00200-00300.trans
  12C-1H4__YT10to10__00300-00400.trans
  12C-1H4__YT10to10__00400-00500.trans
  12C-1H4__YT10to10__00500-00600.trans
  12C-1H4__YT10to10__00600-00700.trans
  12C-1H4__YT10to10__00700-00800.trans
  12C-1H4__YT10to10__00800-00900.trans
  12C-1H4__YT10to10__00900-01000.trans
end
```

Here the keyword abundance is to specify the molecular abundance, which is the factor used in intensities or cross-sections.

3. Cross sections

3.1. Type of spectra: absorption, emission and 'lifetimes'

Only one of these three types is needed, it can appear outside any sections anywhere in the program. To specify the type of spectra, absorption or emission, the keywords `Absorption` or `Emission` are used anywhere in the output file (but outside any section), see examples above.

The lifetimes are computed using the keyword `Lifetime` and treated by ExoCross as a special type of spectra. The lifetime (seconds) of a state j is given by $\tau_j = 1/\sum_i A_{ji}$. As an output, ExoCross will re-print the `.states` file `output.life` with the lifetimes appearing as the 5th column, see Tennyson et al. [2]. Here `output` is the filename specified by the keyword `output`. For example, to compute lifetimes for SiO using the EBJT line list [3], the required input is very short (see `CaH_lifetime.inp`):

```
Lifetime

output 40Ca-1H__Yadin

States 40Ca-1H__Yadin.states
Transitions 40Ca-1H__Yadin.trans
```

The output has the following structure (`40Ca-1H__Yadin.life`):

```
48  2490.086522    8   1.5   5.8179E-03    -    2   1   e
49  2481.998617    4   0.5   5.4309E-03    +    2   0   e
50  1268.411091    8   1.5   1.0865E-02    -    1   1   e
51  1260.127494    4   0.5   1.0143E-02    +    1   0   e
52     8.478269    8   1.5   1.7248E+03    -    0   1   e
53     0.000000    4   0.5  -1.0000E+00    +    0   0   e
54 15256.114138   20   4.5   1.0091E-02    -   18   5   f
55 15256.131981   24   5.5   1.0249E-02    -   18   5   e
```

where the 5th column is with the life times (seconds) and $-1.0000E+00$ is for the states with infinitely long or undefined lifetimes. It should be noted, if the line list contains states with $\tilde{E}'$ higher than the lower energy threshold, the life times make no sense and should be ignored. To the end one can add the keyword `Enermax` specifying the energy threshold $\tilde{E}_{\max}$ (cm^{-1}), which will exclude states with $\tilde{E} > \tilde{E}_{\max}$. The corresponding lifetimes will be also set to $-1.0000E+00$.

3.2. States and Transitions

In order to specify the states file the keyword `States` is used (with aliases StatesFile `StatesFile` and `States_file`), which is followed by the name of the `.states` file:

```
States 12C-1H4__YT10to10.states
```

4

The transition files are specified by the keyword `Transitions` (or `TransitionFiles`) either followed by the name of the `.trans` file, or as part of the `Transitions` section, in case of multiple entries. For example, a single `.trans` can be specified by

```
Transitions 12C-1H4__YT10to10__09400-09500.trans
```

while to specify a set of `.trans` file, use

```
Transitions
 12C-1H4__YT10to10__09100-09200.trans
 12C-1H4__YT10to10__09200-09300.trans
 12C-1H4__YT10to10__09300-09400.trans
 12C-1H4__YT10to10__09400-09500.trans
end
```

3.3. Absorption cross sections using a Doppler line profile

In order to trigger the cross section calculation, the corresponding line profile types need to be specified, for example `Gaussian`, `Doppler`, `Voigt` etc. Here is an example to compute Doppler-type cross sections using an ScH line list.

```
Temperature 1500.0 (K)
Range 0.  16000.0 (cm-1)
Npoints 16001

absorption (type)
Doppler    (line profile)

mass 15.0 (Dalton)

output ScH_1500K_Doppler

States      45Sc-1H__LYT.states
Transitions  45Sc-1H__LYT.trans
```

The Doppler line profile depends on the mass of the molecule, which has to be specified using the keyword `Mass` in Dalton, as in this example. Even an approximate value will suffice for this purpose.

3.4. Absorption cross sections using a Gaussian line profile

A generic Gaussian line profile can be also specified as follows (`VO_gauss_2000K.inp`):

```
Temperature 2000.0 (K)
Range 0.  35000.0 (cm-1)

absorption (type)
Gaussian    (line profile)

hwhm 0.1 (half-width at half-maximum cm-1)

offset 25 (cm-1)

output VO_1500K

States      51V-16O__VOMYT.states
Transitions  51V-16O__VOMYT__30000-35000.trans
```

Here the half-width at half-maximum (HWHM) is given by the keyword `HWHM`. Example inputs are also provided as part of the supplementary material.

3.5. Line profiles

The currently implemented line profiles are

`Gaussian` (`Gauss`) which preserves the area under the line profile;

`Gauss0` Better keyword? is a Gaussian profile computed using the standard sampling approach (does not preserve the line profile area);

`Doppler` (`Doppl`) which preserves the line profile area;

`Doppl0` is the Doppler profile using the sampling approach (does not preserve the line profile area);

`Lorentzian` (`Lorentz`);

`Voigt` is the Voigt profile computed using Humlíček's algorithm [4], which is a sampling approach and does not preserve the area of the line;

`Voigt-Quad` is the Voigt-profile with the area preserved; the the number of quadrature points must be also specified using the keyword `nquad`.

`Voi-Fast` is the Voigt profile computed using the method given in section x.X on the ExoCross paper. On average, 12-13x faster than Humlíček's algorithm; does not preserve the area of the line profile;

`Voi-Fnorm` same as `Voi-Fast` but the area under the line profile is preserved using a simple normalization;

`Bin` is used to compute histograms;

`Box` is the box-type line profile;

`Max` gives the maximal intensity within a given frequency bin.

Here is an input example for cross-sections using the Voigt profile with the conserving area as generated by the quadrature integration method (`Quad-quad`) for NaH (`NaH_Voigt-quad_298K.inp`).

```
Range 0.0   25000.0
Npoints 25001

Absorption
pressure  1.0
Temperature 298.0
mass 23.

species
 H2    gamma 0.0155 n 0.41 t0 298.0 ratio 0.82
 He   gamma 0.1070 n 0.77 t0 298.0 ratio 0.18
end
```

6

```
Voigt-quad
Nquad 40

Output NaH_Voigt-quad_298K_P1

States 23Na-1H__Rivlin.states
Transitions 23Na-1H__Rivlin.trans
```

Here `Nquad` is the number of quadrature points used for Voigt-Quad. For the standard Voigt (Humlíček) using the sampling algorithm the keyword `Voigt` would be used.

3.6. Temperature and pressure

In order to specify temperature and pressure values, the keywords `Temperature` (`Temp`) and `Pressure` are used, respectively. The units are K for the temperature. For pressure any units can be used, as long as the pressure value is consistent with the units of the broadening parameter P_0 specified in the `Species` section (see NaH example above).

3.7. Stick spectra

A stick spectrum (`Stick`) is a list of frequencies and line intensities, accompanied by the full description (quantum numbers) of the upper and lower states. When plotted, each line is represented by a 'stick' with the intensity given by its height. An additional file `output` in the two-column format containing the frequency wavenumbers and line intensities is also created for plotting purposes.

Here is an example, which will produce a 298 K stick spectrum for the NO molecule as part of the the standard output as file as of the file given by the `output` keyword with an extension `.stick`. Each stick line will contain the frequency wavenumber, intensity, upper and lower term values, upper and lower *J* values and all accompanying quantum numbers (upper and lower) as they appear in the `.states` file. The `.stick` file will contain the frequency wavenumbers and intensities only, for example (`CaH_stick_1e-25_298K.inp`):

```
Temperature 296
Range 0.0   5000.0

absorption
stick
threshold 1e-25


output 40Ca-1H__Yadin_296K

States 40Ca-1H__Yadin.states
Transitions 40Ca-1H__Yadin.trans
```

The corresponding standard output will contain the stick line list in the following structure (see Section **??** xxx in the paper for explanation):

```
8.08790500E+00  1.12642733E-25  1.5  2490.0865 <-  0.5  2481.9986  -  2  1  e <-  +  2  0  e
8.28359700E+00  4.06251016E-23  1.5  1268.4111 <-  0.5  1260.1275  -  1  1  e <-  +  1  0  e
8.47826900E+00  1.75317450E-20  1.5     8.4783 <-  0.5     0.0000  -  0  1  e <-  +  0  0  e
1.61109390E+01  2.55982579E-25  1.5  2506.1370 <-  0.5  2490.0261  +  2  2  f <-  -  2  1  f
1.61512000E+01  6.89163546E-25  2.5  2506.2377 <-  1.5  2490.0865  +  2  2  e <-  -  2  1  e
1.65000920E+01  9.22007610E-23  1.5  1284.8485 <-  0.5  1268.3484  +  1  2  f <-  -  1  1  f
1.65418420E+01  2.48211326E-22  2.5  1284.9529 <-  1.5  1268.4111  +  1  2  e <-  -  1  1  e
1.68871990E+01  3.97369277E-20  1.5    25.3005 <-  0.5     8.4133  +  0  2  f <-  -  0  1  f
1.69304360E+01  1.06968718E-19  2.5    25.4087 <-  1.5     8.4783  +  0  2  e <-  -  0  1  e
2.41654150E+01  1.00711872E-24  2.5  2530.3024 <-  1.5  2506.1370  -  2  3  f <-  +  2  2  f
2.42056120E+01  1.91402994E-24  3.5  2530.4433 <-  2.5  2506.2377  -  2  3  e <-  +  2  2  e
........
........
```

while the `output` file `NaH_298K_1e-25.stick` will be in the two columns format containing the frequencies and intensities only. If the line broadening is defined using `Species`, the total line width will also appear in the standard output as the column 8.

3.8. Partition function

The partition function is required for most of the ExoCross calculations. It can be provided using the following three options:

1. Computed using the energies and total degeneracies from the `.states` file, if none of the above is given in the input file (default).

2. Using the keywords `PF` or `Qstat`;

3. From the partition function file `pffile` (To be implemented).

For example:

```
Temperature 298
Range 0 8000.0

PF 1200
....
....
```

Partition function can also be generated on a grid of temperatures, see Section 5.3.

3.9. The frequency (wavenumber or wavelength) range

To specify the frequency range the keywords `Range` or `Wavenumber` are used. By default the frequency is given in wavenumbers (cm^{-1}), which can be changed to the microns using the `um` keyword (not all profiles are currently supported with um). The keyword `cm-1` can be also given after the range values, although with no effect. Here are examples for the range given in wavenumbers:

```
Range 0.0 2000.000 (cm-1)
```

or microns:

```
Range 1.0 1000.0 um
...
```

Here (cm-1) in parenthesis is a description text ignored by ExoCross. In some cases such as when using microns (um) or in combination with the grid defined by the resolving power (R), the range should not start from zero or to be too close to it.

3.10. Broadening parameters using Species

The broadeners in ExoCross are specified using the section Species (alias Broadener) according with the standard scheme given by, where the Lorentzian line width γ_L is

$$\gamma_L = \sum_i \gamma_0^{(0)} \left(\frac{T_0^{(i)}}{T} \right)_i^n \frac{P}{P_0^{(i)}} \rho_i,$$

where $P_0^{(i)}$ and $T_0^{(i)}$ are the reference values of the temperature (K) and pressure, respectively, $\gamma_0^{(i)}$ is HWHM (cm^{-1}) at $T = T_0$ (usually 298 K, which is the default value) for the ith broadener, n_i is the corresponding temperature exponent and ρ_i is the mixing ratio of the broadener $(0 \ldots 1)$. The units of P_0 and the pressure defined by Pressure should be the same.

Here is the list of keywords used in the Species section:

gamma is to define the γ_0;

n is to define the exponent n_i;

delta is to define the pressure line shift shift in cm^{-1} (currently only used to prepare the HITRAN-output);

T0 is to define the reference T (K), usually 298;

P0 is to define the reference pressure, usually 1, the units are the same as for Pressure;

Ratio is the mixing ratio of the species (unitless) for example the solar mixing ratio of H_2 and He is 0.9 and 0.1;

File is the name of the file with broadening parameters.

The name of the species should appear as the first field (string) on a line. Example (NO_1000K_Voigt_P10bar.inp):

```
Temperature 1000.0 (K)
Range 0.   10000.0 (cm-1)

Npoints 10001

absorption
voigt

pressure  10.0 (bar)

mass 40.

offset 25.0

species
 air   gamma 0.045  n 0.70 t0 296.0 file 14N-16O__air.broad  delta 0.000
end

Output 14N-16O_SL_1000K_Voigt_P10

States       14N-16O__NOname.states
Transitions  14N-16O__NOname.trans
```

or

```
Species
  H2  gamma 0.0207 n 0.44 t0 298.0 file 1H2-16O__H2__a1.broad model JJ ratio 0.84
  He  gamma 0.043 n 0.02 t0 298.0 file  1H2-16O__He__a1.broad model JJ ratio 0.16
end
```

4. HITRAN

4.1. Using HITRAN line lists

ExoCross can be used with the HITRAN (HITEMP) line lists [5, 6] (.par files). To do this the keyword HITRAN needs to be put anywhere in the input file (outside any section). This option will also require the partition function (pf) and isotopologue number (Iso) to be specified. Currently only one isotopologue type in time can be processed by ExoCross.

ExoCross uses the HITRAN broadening parameters are unless the species-section is given with user-defined broadening scheme is provide. The HITRAN total statistical weights are also used directly. A .states file is not required and the keyword States is ignored if given.

Here are example for HITRAN's NO line list:

```
Range 0.   8000.0
temperature 1500

absorption

HITRAN
iso 81

pf 8.42423042E+03

stick
threshold 1e-36
```

```
output NO_1500K

Transitions  08_HITEMP2010.par
```

as well as for water

```
Range 0.  10000.0
temperature 1000

absorption

hitran
iso 11

pf 1218.298537

stick
threshold 1e-36

output H2O_1000K

Transitions  08_HITEMP2010.par
```

4.2. Converting to the HITRAN format using the stick option

Another useful functionality of ExoCross associated with HITRAN is to convert an ExoMol line list to the HITRAN format. This option is invoked by (i) the `stick` and (ii) `HITRAN write` keywords. The keyword `write` can optionally also appear inside the `HITRAN` section. Here is an example of converting an HCN ExoMol line list to the HITRAN format (`HCN_HITRAN_296K_convert.inp`):

```
Temperature 296.0
Range 0.  10000.0

iso 231
pressure  1.0

HITRAN write

species
 H2 gamma 0.0155 n 0.41 t0 296.0 file 1H-12C-14N__H2.broad  delta 0.000
 He gamma 0.1070 n 0.77 t0 296.0 file 1H-12C-14N__He.broad  delta 0.000
end

output HCN_HITRAN_296K

abundance 0.97

States 1H-12C-14N__Harris.states
Transitions 1H-12C-14N__Harris.trans
```

The broadening files `1H-12C-14N__H2.broad` and `1H-12C-14N__He.broad` are given as part of the supplementary and can be obtained from www.exomol.com. The line list files `.states` and `.trans` can be downloaded from www.exomol.com. HITRAN's `ID_ISO` is specified using the `Iso` keyword as before.

The air-broadening parameters of NO in the example above are defined in `NO_broad.dat`.

The HITRAN's error codes can be given as part of the `HITRAN` section as in the following example of converting ExoMol's NO line list (`NO_to_HITRAN_296K.inp`):

```
Temperature 296.0 (K)
Range 0.  30000.0 (cm-1)

iso 81
pressure  1.0

HITRAN write

species
 air    gamma 0.045  n 0.70 t0 296.0 file 14N-16O__air.broad  delta 0.000
end


 HITRAN  write
  error-E  qn 10 ierr 4  vmax  10 ierr 3  vmax  20 ierr 2  vmax  30  ierr 1 vmax 100
  error-S  qn 10 ierr 5  vmax  10 ierr 4  vmax  20 ierr 3  vmax  100
  error-Air   ierr 4
  error-self  ierr 4
  error-n     ierr 4
  error-delta ierr 0
 end

Output 14N-16O__NOname

 States      14N-16O__NOname.states
 Transitions 14N-16O__NOname.trans
```

In this case the error codes are distinguished using the vmax keywords in the error-E and error-S structures: vmax identifies the range of the quantum number from the qn = 10 column for the error ierr. The error-keywords error-Air, error-self, error-n and error-delta are currently single-valued. The line list will be written into the file 14N-16O__NOname.par.

To summarize the ExoCross keywords specifying the HITRAN error codes:

error-E stands for the error code related to the energy value; the line includes ranges defining the errors;

error-S stands for the error code related to the absorption intensity (line strength) value; the line includes ranges defining the errors;

QN is the column number with the quantum number used for the error-specifications (the state ID is the column number 1);

ierr is followed by the error;

vmax is followed by the maximum value of QN, this error code applies to;

error-Air is to specify the error codes for the HITRAN air-broadening (single value);

error-self is to specify the error code for the HITRAN self-broadening (single value);

`error-n` is to specify the error code for the HITRAN *n*-broadening (single value);

`error-delta` is to specify the error code for the HITRAN's line-shift δ (single value).

Another set of fields required by HITRAN is the source indices. Currently these fields are set to 1. More options will be introduced in the future. The data sources? We need to introduce it or explain.

delta is still not specified?

Here is simplest example, where ExoCross uses some default broadening and error values to produce a HITRAN-type format:

```
Temperature 296.0 (K)
Range 0.  30000.0 (cm-1)

HITRAN write

Output 14N-16O__NOname_simple

States       14N-16O__NOname.states
Transitions  14N-16O__NOname.trans
```

4.3. Histograms

ExoCross can compute cross sections from pre-computed binned histograms (produced using the `bin`-keyword) treating them as lists of artificial lines (super-lines) with pre-computed (absorption) coefficients at a given T [7]. The advantage of this approach is that calculations of cross sections from such a structure are much faster due to the smaller number of bins (super-lines) relative to the number of standard transition lines. The approximation used is that the line-centres for all transitions within a given bin sifted to the centre of the bin.

Super-lines can be produced by ExoCross using the keyword `Bin` as a 'profile'-type, as histograms of intensities (total absorption/emisssion) per each frequency bin. Super-lines are written into an external file with the name specified with `output`. The histograms can be also computed on a non-uniform grid, for example, on a grid of a constant resolving power (keywords `R` or `Resolving`). It should be noted that $R = \Delta\tilde{\nu}/\tilde{\nu}$ are ill-defined small values of $\tilde{\nu} = 0$ and therefore the wavenumber range must start from some sensible positive value. In the example below a binned Histogram is generated for the NO molecules at $R = 1,000,000$ and $T = 1000$ K, where the grid starts at 10 cm^{-1}(`NO_1000K_bin.inp`):

```
Temperature 1000.0 (K)
Range 0.  30000.0 (cm-1)

R 1000000

Npoints 10001

absorption
bin
```

```
Output 14N-16O__NOname_1000K_R100K

States       14N-16O__NOname.states
Transitions  14N-16O__NOname.trans
```

The histogram here are written to `H2O_bin_T296K_R1M.xsec` as a two-column output $(\tilde{\nu}_{fi}, I_{fi}(T))$, where $I_{fi}(T)$ is a line intensity (absorption coefficient) at the temperature T. Please note that the standard extension used for super-lines is `.super`, and therefore the file is renamed to `14N-16O__NOname_1000K.super`.

In order to use histogram, the keyword `Histogram` (alias `Super-line`) needs to be place anywhere in the body of the input file. The `Transition` keyword now has to refer to the histogram file name (`14N-16O__NOname_1000K_R100K.super` in this example). The `.states` file is not needed, otherwise the input is as usual:

```
Temperature 1000.0 (K)
Range 0.  10000.0 (cm-1)

Npoints 10001

super-lines

absorption
gaussian

Output 14N-16O_SL_1000K

States       14N-16O__NOname.states
Transitions  14N-16O__NOname_1000K_R100K.super (binned-histograms)
```

It should be noted that the super-lines cannot take into account state-dependent line broadening, therefore when requesting for Voigt or Lorentzian, only the state-independent γ_0 and n parameters will be used. Different broadeners (`Species`) can be still mixed as shown in the example below (`NO_1000K_super-lines_Voigt.inp`):

```
Temperature 1000.0 (K)
Range 0.  10000.0 (cm-1)

Npoints 10001

super-lines

absorption
voigt

pressure  10.0 (bar)

mass 40.

offset 25.0

species
 air   gamma 0.045  n 0.70 t0 296.0 delta 0.000
end


Output 14N-16O_SL_1000K_Voigt_P10

States       14N-16O__NOname.states
Transitions  14N-16O__NOname_1000K_R100K.super
```

5. Other functionality

5.1. Applying filters

Specific (electronic, vibrational, or vibronic) bands, specific branch, or even specific transitions based on a selection of quantum number can be selected using the the `Filter` section-type structure. This structure is based on the two main keywords, `Upper` and `Lower`, followed by the column number as appear in the `.states` file and the corresponding value (quantum label) to be used in selection. The columns are counted starting from the very first one, i.e. the ID column is the column number 1. Currently one can list up to 100 filters. This maximum value however can be changed in the code by modifying the parameter `Filtermax`.

The filter section has the following structure

```
Upper col1 Label1 Lower col2 Label2
Upper col3 Label3
Lower col4 Label4
Lower col5 Label5
Upper col6 Label6
Lower col7 Label7
.....
```

where `Upper` and `Lower` refer to the upper and lower states, respectively; label is the reference quantum label (number or string) which appears in the `col`-column (integer, 1,2,3...). Below is an example of an input file to do an NaH stick spectrum, where only the states labelled `A` (upper) and `X` (lower) are selected. The corresponding labels appear in the column 6 of the corresponding `.states` file (`NaH_stick_1e-25_298K_filter.inp`).

```
Temperature 296
Range 0.0  25000.0

absorption
stick
threshold 1e-28

filter
 lower  6  X   upper  6  A
end

Output NaH_298K_1e-25_filter

States 23Na-1H__Rivlin.states
Transitions 23Na-1H__Rivlin.trans
```

The standard output will have the following structure:

```
1.62259740E+04  1.02336295E-28   6.0   22691.7508 <-    7.0    6465.7767  2.7131e-08   A   1 <- 6.9327e-03   X
1.62605007E+04  1.15438339E-28   5.0   22670.0544 <-    6.0    6409.5537  2.7031e-08   A   1 <- 6.8829e-03   X
1.62799349E+04  1.14715559E-28   8.0   22745.7116 <-    7.0    6465.7767  2.7379e-08   A   1 <- 6.9327e-03   X
1.62906595E+04  1.21763600E-28   4.0   22651.9242 <-    5.0    6361.2648  2.6948e-08   A   1 <- 6.8369e-03   X
..............
```

Here is an example of a multiple filters section (CH_4 line list):

```
filter
  upper  6 0 lower  6 0
  upper  7 0 lower  7 0
  upper  8 0 lower  8 0
  upper  9 0 lower  9 0
  upper 10 0 lower 10 0
  upper 11 0 lower 11 0
  upper 12 1 lower 12 0
  upper 13 1 lower 13 0
  upper 14 1 lower 14 0
end
```

In this case this is to select Methane's vibrational band ν_4 (0 00 000 111 ← 0 00 000 000).

5.2. Cooling function

The cooling function $W(T)$ is computed as emissivity (erg/molecule/sterradian) on a grid of temperatures (K). Here is an example of the cooling calculations using the NO line list for temperatures up to 2000 K with a step of 10 K (see `NO_cooling_2000K.inp`):

```
Range 0.0  10000.0

emission

cooling
 tempmax 2000
 ntemps  200
end

output 14N16O_2000K

States      14N-16O__NOname.states
Transitions 14N-16O__NOname.trans
```

This calculation will produce a two-column output ($W(T)$ vs T) written into a `.cooling` file (14N16O_2000K) as well as a partition function file `.pf` in the same structure (see below).

The main keywords are:

`Ntemps` is the number of temperature steps. Alias is `Npoints`.

`Tempmax` is the maximal temperature in K (minimal $T = 1$ K); `Maxtemp` or `Max-Temperature` are aliases.

5.3. Partition function and specific heat

For the partition function $Q(T)$ and specific heat (heat capacity) $C_p(T)$ is computed using a similar to the `cooling` structure. For example, to evaluate partition function values of NO using the states from `14N-16O__NOname.states` on a grid of $\Delta T = 10$ K up ro $T = 2000$ K, the ExoCross input has the following form (`NO_pf.inp`):

16

```
partfunc
 tempmax 2000
 ntemps  200
end

output 14N16O_2000K

States      14N-16O__NOname.states
```

The `.trans` file is not needed. A two-column output will be written into `14N16O_2000K`.

Here is amother example of the `Partfunc` section, where the specific heat of C_p for NO is requested:

```
partfunc
 tempmax 2000
 ntemps  200
 Cp
end
```

Apart from $Q(T)$ and $C_p(T)$, the 1st and 2nd moments (see Vidler and Tennyson [8]) can be also computed using the keywords `Moment` followed by 0 (none), 1 (Q_1), 2 (Q_2) or 3 (C_p) e.g.:

```
partfunc
 Ntemps     500
 tempmax  5000.0
 Moment 1
end
```

Apart from the actual values of these quantities, ExoCross will also print out the intermediate values of $Q(T)$ for each J and T, which is useful for studying the convergence of the line list at different T, see, for example, Sousa-Silva et al. [9]. However for this option the states in the `.states` file must be sorted by energies.

The main keywords used in the `Partfunc` structure are

`Ntemp` (or Npoints): number of temperature steps.

`tempmax`: maximal temperature in K (minimal $T = 1$ K); `Maxtemp` or `Max-Temperature` are aliases.

`Moment` (0,1,2, or 3): moment to compute, 0 is the partition function (default and can be omitted), 1 is the 1st moment, 2 is the 2nd moment and 3 is CP.

`Cp` is equivalent to Moment 3.

5.4. Intensity thresholds

The intensity threshold is used to reduce the output in stick-type spectra calculations. The main keyword is `Threshold` followed by the intensity value (cm/molecule). A line is skipped from intensity processing if the corresponding absorption coefficient/emissivity is smaller than the threshold:

```
Threshold 1e-28
```

A more sophisticated method is to use the dynamic HITRAN's intensity cut-off, defined by the ν_{crit} and I_{crit}, with default values are 2000 cm^{-1} and 10^{-29} cm/molecule, respectively.

Here is an example of the input with a dynamics HITRAN cut-off for NaH (`NaH_stick_1e-29_296K_HITRAN_cutoff.inp`):

```
Temperature 296
Range 0.0  30000.0

absorption
stick

cutoff HITRAN (I_crit) 1e-29  nu_crit 2000

Output NaH_298K_1e-25_Icrit

States 23Na-1H__Rivlin.states
Transitions 23Na-1H__Rivlin.trans
```

5.5. Treating non-LTE

In order to treat non-LTE the temperature can be split into rotational temperature and vibrational temperature, which in the input is given as

```
Temperature  (Rotational) 700 Vib (vibrational) 2000 (K)
```

In this case it is also important to provide the addresses of the vibrational columns using the `QN` (quantum numbers) section, as shown below.

5.6. Quantum numbers (`QN`)

The designation of the quantum numbers (QN) to columns can be done using the `QN` section. In this section `vib` is used for the range of the vibrational QNs; `K` is for the column with the rotational K; `Sym` is for symmetry; `Nsym` is number of the symmetries; `Nmodes` stands for the number of vibrational modes. The column numbering starts after the J-column (column number 4). The vibrational addressing is required for when vibrational temperature is used. Here is an example of non-LTE cross-sections (`NO_non-LTE_700-2000K.inp`) to simulate cross sections of NO at $T_{rot} = 700$ K and $T_{vib} = 2000$ K:

```
Temperature  (Rotational) 700 Vib (vibrational) 2000 (K)
Range 0.0  40000.0

absorption
gaussian 0.2

QN
  Vib 10 10
  Nmodes 1
end

output 14N16O_non-LTE

States      14N-16O__NOname.states
Transitions 14N-16O__NOname.trans
```

The vibrational quantum numbers are listed in the 10th column (`Vib 10 10`), which are used to estimate the vibrational energy contribution.

5.7. Memory, parallelization and vectorization

In order to facilitate the parallelization and vectorization of the intensity calculations, ExoCross process the transition lines by chunks. The number of lines to read ('to cache') and then to process simultaneously is specified by the keywords `Ncache` (aliases are `NRAM` or `LINES-TO-CASH`). If this value is not given, it will be estimated based on the amount of the memory available M_{max} on the system, which is specified by the keywords `Mem` (alias `Memory`), followed by the keywords `Gb` (default) or `Mb`. The default value of `Ncache` is 100. The `mem` value is the value available on the system for the ExoCross job. The memory requirements can be estimated using the following expression

$$M = N_{\text{Cache}}(6 \times 8 + 5 \times 4)/1024^3 \quad \text{Mbyte}$$

where 6 and 5 the number of 1D (real(8)) and `integer(4)` matrices with N_{Cache} elements for different ExoCross objects. Therefore, if `Ncache` is not specified, the number of lines to cache N_{cashe} is estimated by

$$N_{\text{Cache}} = \frac{M_{\text{curr}}}{(6 \times 8 + 5 \times 4)} 1024^3$$

where M_{curr} is the current available memory (M_{max} minus used by ExoCross) in Mbyte.

Additionally the, number of the OpenMP processes should be specified by the keyword `Nprocs`. The default value is 1 Check if it can be increased. Ideally it should be the same as the number of OpenMP-processors allocated, but would work with any number. Aliases are `OMP_NUM_PROCS` and `OMP_PROCS`. Example:

```
mem 64 Gb

Ncache 10000000
Nprocs 16

Temperature 1000
Range 0.0  10000.0

emission
gaussian
hwhm 0.1

output 14N16O_2000K

Output 14N-16O_emiss_g.1

States       14N-16O__NOname.states
Transitions  14N-16O__NOname.trans
```

19

6. List of keywords

`Absorption`: defines type of spectra. This is a single-keyword input, no additional values are required.

`Abundance`: defines the molecular abundance (≤ 1 and ≥ 0), which is a factor used in intensities or cross-sections.

`Box`: Line profile type, single-keyword input.

`Bin`: Line profile type, single-keyword input.

`cm-1`: units of wavenumber used in combination with `Range`, can be omitted as default.

`Cooling`: section name to compute the cooling function on a grid of temperatures. The parameters used in the Cooling section are:

  – `Ntemps` (`Number-of-temperatures`): Number of temperature steps (integer, ≥ 1).

  – `TempMax` (`Max-Temperature`): Maximal temperature in Kelvin (> 0).

`Cutoff HITRAN`: section to define HITRAN cutoff s_{crit} parameters, followed by

  – `nu_crit`: a critical wavenumber frequency to specify v_{crit} (cm^{-1}) as part of the HITRAN cut-off scheme.

`Doppler` (`Doppl`, `Doppl0`): profile type, single-keyword input.

`Emission`: type of spectra, single-keyword input.

`Enermax`: Energy threshold (> 0) used to select states below some energy value (cm^{-1}) (or to skip states from EXOCROSS calculations above some energy term value). The default value is 1,000,000.

`Filter` (`Select`): The filter section to specify how to select particular bands or transitions for EXOCROSS processing. The `Filter`-specific parameters are:

  – `Upper`: the column number followed by the upper state quantum label (treated as a string of characters of arbitrary length).

  – `Lower`: the column number followed by the lower state quantum label (arbitrary string).

`Gaussian` (`Gauss`): profile type, single-keyword input.

`Gauss0`: profile type (single-keyword input) sampling method (the line area is not conserved).

`GF`: to compute the oscillator strength gf as part of converting to the PHOENIX format.

`GF_factor`: defines the correction factor (real, > 0) to scale down/up the oscillator strength `GF` when converting to the PHOENIX format due to the different conventions used to treat the oscillator strength (in Phoenix convention the individual statistical weights are ignored).

`Histogram (super-line,super-lines)`: is to compute cross sections from histograms (super-lines), usually generated using `bin`. Single-keyword.

`HITRAN`: to use the HITRAN-format line list (`.par`). it requires also a definition of the partition function `pf` and of the isotopologue number `iso`. The `.states` file is not required. Single-keyword input.

`HITRAN write`, section: to convert ExoMol to the HITARN format. `iso`, `species`, `Temperature` are required (More?). The section is used to define the HITRAN error-codes:

- `error-E` and `error-S`: used to specify the ranges for the quantum numbers for different Energy and intensity error-codes, respectively. Appear a the beginning of a line.
- `qn`: is to specify column number (integer) containing quantum numbers used for the error-specification.
- `ierr`: is followed by the error code (integer), which is followed by by `vmax` followed by the maximum value of `qn` this error code applies to, which is followed by another error code.
- `vmax`: maximal value of `qn` for a given error-code (integer).
- `error-Ai`: to specify the error code for the HITRAN air-broadening (single constant integer value).
- `error-ierr`: to specify the error code for the HITRAN self-broadening (single constant integer value).
- `error-n`: to specify the error code for the HITRAN n-broadening parameter (single constant integer value).
- `error-delta`: to specify the error code for the HITRAN line-shift (single constant integer value).

HWHM (`HalfWidth`): Half width at the half maximum (real), used for `Gauss` and `Lorenz` types of line profiles. It will also define γ_0 for Voigt if the `Species` section is not given.

`Ioffset`: an integer value to compute the wavenumber offset in units of HWHM: `offset = ioffset`×`hwhm`.

`Iso` (`isotope`): isotopologue number of a molecule (integer > 0) in HITRAN (e.g. 261 for $^{12}C_2H_2$) or Phoenix classification (e.g. 192 for $H_2^{16}O$).

`Lifetime`: to compute life time of E_j as $1/\sum_i A_j i$ (single-keyword).

`Mass` (`Masses`): Effective molecular mass (Dalton), used for Doppler and Voigt profiles.

`Mem` (`Memory`): maximal memory allocated for the job (Mb or Gb). It is used to estimate how many transitions can be put into RAM. Should be less than or equal too the memory of the system (real or integer).

`Mb` and `Gb`: used to specify the units of `Memory`.

`Nprocs` (`OMP_NUM_PROCS`, `OMP_PROCS`): Number of OpenMP processes (integer¿0). Ideally should be the same as the number of OpenMP-processors allocated, but would work with any number.

`Ncache` (`Lines-to-Cache`, `NRAN`): number of transitions to put into the memory (integer, >0). The default value is 100.

`Npoints` (`Number-of-points`): Number of cross-section grid points (integer, usually an odd number, > 0). The default is 1001.

`Nquad`: Number of quadrature points used for `Voi-Quad` (integer > 0). The default value is 20.

`Offset`: a wavenumber offset in cm^{-1} in cross section calculations (real or integer). The default is 25.0. In line profile calculations the frequency offset specifying the calculation region around each line plays a special role for pressure broadened cross sections. Not only does it influence the calculation time and the accuracy of cross sections, but it also used in some cases as part of conventions. For example the water cross section simulations are conventionally done with the 25 cm^{-1} cutoff.

`Output`: File name for cross sections, stick-spectra etc, lifetimes, cooling etc (string $\leq$ 80 characters).

`Partfunc` (Partition-function): Section to compute the partition function and its contributions for a set of temperature. The parameters used in the `partfunc` section are:

- `Ntemps` (Number-of-temperatures): Number of temperature steps (integer > 0).
- `TempMax` (Max-Temperature): Maximal temperature (integer or real, > 0).
- `Moment`: a moment to compute (0=none,1=Q_1,2=Q_2,3=C_p); it is used as part of the partition function (`partfunc`) section.
- `CP`: To compute specific heat, the same as `Moment 3`, used as part of the partition function (`Partfunc`) section.

`Phoenix`: to convert ExoMol to the Phoenix formats. The section `Species` is also required to specify the broadening parameters. `Phoenix` should appear anywhere in the main body of the input. Single-word. This is very similar to the `HITRAN` option, only using different formatting approach. under construction. The keyword `Phoenix` should appear anywhere in the main body of the input and outside any section.

`PF` (QSTAT): external partition function value for the given temperature (integer or real). If not given, the partition function is computed using energies from the `.states` file.

`Pressure`: The pressure value (the same units as for P_0 (P0) in `Species` (real or integer, > 0).

`QN` (quntum-numbers): section to associate quantum numbers with columns as they appear in the `.states` file. The vibrational QNs are required for non-LTE calculations, where the vibrational temperature is used. The J number is always column 4. The `QN`-specific parameters are

- `Nmodes`: the number vibrational modes (integer).
- `vib`: two integer numbers specifying the range of the columns with the vibrational QNs;
- `K`: the column number with the rotational K (integer);
- `Sym`: the column number containing the symmetry (integer);
- `Nsym`: the number of the symmetries (integer).

`Rect`: Profile type, single-word.

`Resolving` (R): Resolving power value used to define a non-equidistant grid (integer > 0).

Range (`Window`): Two numbers ν_A and ν_B specifying the wavenumber (cm^{-1}, default) or wavelength (νm in combination with um) range $[\nu_A, \nu_B]$. In combination with `Resolving` or um, ν_A must be large than zero.

Species (`Broadener`): Section to define the Voigt-type broadening parameters. The specific keywords used in `Species` are

- n: Temperature exponent n_i (unitless, > 0);
- Gamma: Reference HWHM (cm^{-1}, > 0);
- Delta: Pressure line shift (cm^{-1}, $\geq$ 0).
- T0: Reference T (K, > 0), default is 296 K.
- P0: Reference pressure ($\geq$ 0), any units can as lons as they are consistent with that used in `Pressure`, default is 1.
- Ratio: Mixing ratio of the broadeners (unitless), for example the solar mixing ratios of H_2 and He are 0.9 and 0.1.
- File: Filename containing the broadening parameters.

States (`StatesFile, StateFile, States_file, States-File`): Name of the .states file (a string of the length $\leq$ 80).

Sticks: "Profile" type, to print intensities as 'sticks' (intensity vs line position) together with the corresponding quantum numbers.

Threshold: Intensity threshold to skip weak lines from any calculation, if the corresponding absorption coefficients/emissivities are smaller than the `Threshold` value (in the units of absorption coefficient or emissivity).

Transitions (`TransitionFiles`): Name of the .trans file or a section name for the list of the .trans files.

Verbose: Verbose level to define the amount of data to be printed out in the output (Default is 2).

micron (um): To switch to the wavelength in μm as part of the keyword `Range` instead of wavenumbers.

Voigt: Profile-type Voigt generated using Humlíček's algorithm, single-word. A sampling method is used, the the line profile is not conserved.

Voi-Quad: Profile-type Voigt generated using the quadrature integration, single-word. The the line profile is conserved.

`Voi-fast`: Profile-type Voigt generated using our new Voigt-fast algorithm, single-word. A sampling method is used, the the line profile is not conserved.

`Voi-norm`: Profile-type Voigt generated using our new Voigt-fast algorithm, single-word. The the line profile is conserved.

6.1. Compilation under Linux

To compile ExoCross use the `make` command on Linux, which will execute the makefile provided with the ExoCross package.

7. Programmer's guide to ExoCross

In this section we describe the structure of the source code. This can be useful to people who plan to modify ExoCross, e.g. to implement new features.

At present ExoCross's source code consists in 5 files for a total of XXX lines of code in total (including comments). The three files containing the bulk of the program are:

1. `crosssecrions.f90` (XXX lines) is a small front-end which calls the subroutines for energy, line intensities or fitting calculations.

2. `spectrum.f90` (xxxx lines) contains the main part of the code.

3. `accuracy.f90` (xxxx lines) contains values of physical constants, variable kind parameters and other data to be shared.

4. `timer.f90` (xxxx lines) contains utility subroutines which monitor the time spent on various parts of the code as well as memory usage.

5. `input.f90` (xxxx lines) is an input parsing module written by Anthony J. Stone, see [1]. Is is used by Duo to read the input file.

6. `VoigtKampff_module.f90,VoigtKampffCollection_module.f90` (xxxx lines) Subroutines defining the Voigt-fast method.

Here below we give some comments on the structure of the code.

The source file `crosssections.f90` is a simple 'driver' which calls the following subroutines:

`accuracyInitialize` from the `accuracy` module (`accuracy.f90`) to initialize different physical constants and parameters

`ReadInput` to read and interpret control parameters and keywords from the input file (`spectrum.f90`).

`Intensity` is to perform all the calculations (`spectrum.f90`).

`MemoryReport` to print out the program memory usage (`timer.f90`).

`TimerReport` to print out the program time usage (`timer.f90`).

The source file `spectrum.f90` contains a module called `spectrum` which provides the bulk of subroutines necessary for running ExoCross. It also contains the subroutine `ReadInput` which deals with reading the inputs file, performing initial consistency checks etc.

7.1. How to program a new line profile

Here we list and explain the main steps required to include a new line profile into to ExoCross. Let this be a user-defined new profile called, for example, **Super-profile**.

1. A string `'SAMPLE-PROFILE'` specifying the corresponding keyword must be added to the following sections, using uppercase:

   Line 633 (`ReadInput` subroutine):

   ```
   case('SUPER-PROFILE','GAUSSIAN','GAUSS','DOPPL','DOPPLER','RECT','BOX','BIN')
   ```

   Line 1155:
   ```
   case ('SUPER-PROFILE','GAUSS','DOPPL','LOREN','GAUSO','DOPPO',....)
   ....
   ```

   Line 2256 (`intensity` subroutine, first three characters):

   ```
   if (any( trim(proftype(1:3))==(/'SUP','DOP','GAU','REC','BIN','BOX',..../)) ) then
   ```

   Line 2326 (first three characters):

   ```
   elseif (any( trim(proftype(1:3))==(/'SUP','DOP','GAU','REC','BIN','BOX',.../)) ) then
   ```

2. The main part of the program (subroutine `intensity` of `spectrum.f90`) where the line profiles are utilized is within the following SELECT CASE structure (starting from the line 1915):

   ```
   select case (trim(proftype(1:5)))
   ....
   ```

   This line profile utilization section should be given in the following form:

```
case ('SUPER')
    !
    !$omp parallel do private(iomp,iswap,abscoef,tranfreq,halfwidth) &
    shared(intens_omp) schedule(dynamic)
    do iomp = 1,N_omp_procs
      !
      do iswap = iomp,nswap,N_omp_procs
        !
        abscoef = abscoef_ram(iswap)
        tranfreq = nu_ram(iswap)
        halfwidth = gamma_ram(iswap)
        !
        call do_superprofile(tranfreq,abscoef,dfreq,freq,halfwidth,offset,freql,intens_omp(:,iomp))
        !
      enddo
      !
    enddo
    !$omp end parallel do
```

where only the first five characters of the name-string ('SUPER') are used
for CASE. Here superprofile is the user-defined subroutine specifying the
functional form of the line profile. It is up to the user to define the order and
the input/ouput arguments of the subroutine, see, for example, the subrou-
tines do_lorentz, do_gauss, do_Voigt etc. In this example, the following
arguments are used:

tranfreq is the line position (transition frequency), either in wavenumber
(default) or micron.

abscoef is the integrated intensity value (absorption coefficient or emissiv-
ity) in corresponding units.

dfreq is the size of the frequency grid in the units of tranfreq.

freq is an array containing values of the frequency (wavenumber of mi-
cron) grid of the dimension npoints.

halfwidth is the value of the line width (as taken from the array gamma_ram).

offset is the frequency offset value.

freql is the value at the left-most frequency grid point.

intens_omp(:,iomp) is an array (out) which will be used to store the grid
of computed cross sections for the given OpenMp thread iomp.

27

Index

[1] A. J. Stone, input.F90 a Fortran90 module for parsing text input, see http://www-stone.ch.cam.ac.uk/programs/, 2005.

[2] J. Tennyson, S. N. Yurchenko, A. F. Al-Refaie, E. J. Barton, K. L. Chubb, P. A. Coles, S. Diamantopoulou, M. N. Gorman, C. Hill, A. Z. Lam, L. Lodi, L. K. McKemmish, Y. Na, A. Owens, O. L. Polyansky, T. Rivlin, C. Sousa-Silva, D. S. Underwood, A. Yachmenev, E. Zak, The ExoMol database: Molecular line lists for exoplanet and other hot atmospheres, Journal of Molecular Spectroscopy (2016) – ISSN 0022-2852, doi:http://dx.doi.org/10.1016/j.jms.2016.05.002, URL http://www.sciencedirect.com/science/article/pii/S0022285216300807.

[3] E. J. Barton, S. N. Yurchenko, J. Tennyson, ExoMol line lists - II. The rovibrational spectrum of SiO, Mon. Not. R. Astron. Soc. 434 (2) (2013) 1469–1475, doi:10.1093/mnras/stt1105.

[4] J. Humlicek, EFFICIENT METHOD FOR EVALUATION OF THE COMPLEX PROBABILITY FUNCTION - VOIGT FUNCTION AND ITS DERIVATIVES, J. Quant. Spectrosc. Radiat. Transf. 21 (1979) 309–313, doi:10.1016/0022-4073(79)90062-1.

[5] L. S. Rothman, I. E. Gordon, R. J. Barber, H. Dothe, R. R. Gamache, A. Goldman, V. I. Perevalov, S. A. Tashkun, J. Tennyson, HITEMP, the high-temperature molecular spectroscopic database, J. Quant. Spectrosc. Radiat. Transf. 111 (2010) 2139–2150, doi:10.1016/j.jqsrt.2010.05.001.

[6] L. S. Rothman, R. B. Wattson, R. R. Gamache, J. Schroeder, A. McCann, HITRAN, HAWKS and HITEMP high-temperature molecular database, Proc. Soc. Photo. Opt. Instrum. Eng. 2471 (1995) 105–111.

[7] S. N. Yurchenko, D. S. Amundsen, J. Tennyson, I. P. Waldmann, A hybrid line list for CH_4 and hot methane continuum , Astron. Astrophys. .

[8] M. Vidler, J. Tennyson, Accurate partition function and thermodynamic data for water, J. Chem. Phys. 113 (2000) 9766–9771.

[9] C. Sousa-Silva, N. Hesketh, S. N. Yurchenko, C. Hill, J. Tennyson, High temperature partition functions and thermodynamic data for ammonia and phosphine, J. Quant. Spectrosc. Radiat. Transf. 142 (2014) 66–74, doi:10.1016/j.jqsrt.2014.03.012.